JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 12 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v12i3.5283

DETECTION OF CYBERBULLYING USING SVM, NAIVE BAYES, AND RANDOM FOREST ALGORITHM

Monica Fachrita Ruziqiana[1], Lailatul Hidayah[2*], Mohammad Arif Rasyidi[3]

[1,2,3] Universitas Internasional Semen Indonesia, Kompleks PT. Semen Indonesia (Persero) Tbk, Jl. Veteran, Kb. Dalem, Sidomoro, Kec. Kebomas, Kabupaten Gresik, Jawa Timur 61122

Received: 30 Juli 2024
Accepted: 3 Agustus 2024
Published: 15 Agustus 2024

Keywords:
Cyberbullying;
SVM;
Naïve Bayes;
Random Forest

Corespondent Email:
lailatul.hidayah@uisi.ac.id

Abstrak. Penggunaan media sosial terus meningkat, dengan Instagram muncul sebagai salah satu platform yang paling menonjol. Pada Januari 2023, Instagram memiliki 1,318 miliar pengguna, yang sebagian besar berusia 18-24 tahun. Meskipun remaja melaporkan peningkatan kepercayaan diri dan berkurangnya rasa kesepian, media sosial juga memfasilitasi perundungan siber yang memengaruhi 35% remaja dengan kesejahteraan emosional yang rendah. Penelitian ini bertujuan untuk mengembangkan model deteksi perundungan siber dalam komentar Instagram, mengklasifikasikannya ke dalam kategori negatif, positif, dan netral menggunakan algoritma SVM, Naïve Bayes, dan Random Forest. Metodologi mencakup pengumpulan data, prapemrosesan, transformasi teks menggunakan TF-IDF, dan analisis komparatif. Grid search digunakan untuk mengoptimalkan parameter algoritma. Hasil awal menunjukkan bahwa Naïve Bayes dan SVM mencapai akurasi sebesar 75,47%, sedangkan Random Forest mencapai 69,88%. Setelah penyetelan parameter, akurasi SVM meningkat menjadi 97,79%, sedangkan Random Forest menurun menjadi 66,51%. Temuan ini menekankan kinerja superior SVM dengan konfigurasi parameter dalam mendeteksi perundungan siber di Instagram.

Abstract. *Social media usage has been steadily increasing, with Instagram emerging as one of the most prominent platforms. As of January 2023, Instagram had 1.318 billion users, predominantly aged 18-24. While teenagers report enhanced self-confidence and diminished feelings of loneliness, social media also facilitates cyberbullying, impacting 35% of adolescents with low emotional well-being. This research seeks to develop a model for detecting cyberbullying in Instagram comments, classifying them into negative, positive, and neutral categories using SVM, Naïve Bayes, and Random Forest algorithms. The methodology encompasses data collection, preprocessing, text transformation via TF-IDF, and a comparative analysis. Grid search is employed to optimize algorithm parameters. Initial results indicated that Naïve Bayes and SVM achieved an accuracy of 75.47%, while Random Forest reached 69.88%. Following parameter tuning, SVM's accuracy improved to 97.79%, whereas Random Forest's decreased to 66.51%. The findings underscore the superior performance of SVM with parameter tuning in detecting Instagram cyberbullying.*

1. INTRODUCTION

In this rapidly changing technological era, social media is advancing as well. One of the

most popular and constantly growing social media is Instagram. According to data published by Facebook, which is Instagram's parent company, Instagram's active users in January 2023 reached 1318 billion users worldwide. That means as many as 21.1% of people aged 13 years and over worldwide are Instagram users. As of January 2024, statistical data shows that Indonesia has 100.9 million Instagram users, ranking it as the fourth-largest Instagram user base worldwide, after India, the United States, and BraziL[1]. Unfortunately, there are still a lot of users who aren't wise in using Instagram. Many users abuse and do not act based on good ethics in socializing via Instagram.

A survey by the anti-bullying organization Ditch The Label reveals that Instagram is the social media platform most often associated with cyberbullying. The results show that more than 42% of cyberbullying victims have been bullied in Instagram media [2]. Cyberbullying in Indonesia is an increasingly prevalent issue that necessitates an immediate response. According to research conducted by ChildFund, nearly 50% of high school and university students have experienced online bullying, with 59% reporting an incident within last quarter of 2023[3].

Cyberbullying means using digital technology like social media or instant messaging to harm someone. This can include abusive messages, gossiping, sharing embarrassing content and excluding individuals from online activities. Unlike traditional bullying that is limited to a certain time or place, cyberbullying can happen any time and quickly reach a wider audience causing greater emotional distress for the victim. With the growing prevalence of social media and other digital platforms, individuals of any age or background can be impacted by this issue.

Cyberbullying can also involve intentional and continuous harassment from peers who hold more power than the victim. Victims of cyberbullying frequently suffer from adverse psychological and emotional impacts, including anxiety, depression, low self-esteem, and, in severe cases, may develop suicidal tendencies [4]. Research by WebMD suggests that individuals who are victims of cyberbullying may encounter persistent emotional, concentration, and behavioral challenges. Victims frequently experience trust issues and are at an increased risk of engaging in alcohol or drug abuse at a younger age[5]. Cyberbullying is a serious problem that can have a negative psychological impact on victims, hence this problem must be addressed immediately, especially in adolescents.

Cyberbullying is associated with significant mental and psychological disturbances among Generation Z, making it a serious public health issue. Many victims of cyberbullying become withdrawn and, among them, a significant number feel insecure about posting activities on social media[6] . According to a previous study conducted by Annisah Rachmawati and Yuli Andrasri in 2022, there are several types of cyberbullying behaviors observed in Instagram comment sections. These include assigning victims derogatory nicknames, using demeaning language towards victims, and threatening the safety of victims [7].

2. LITERATURE STUDY

The problem of cyberbullying on Instagram comments is an important and interesting thing to study as text data processing. There has been numerous research around the detection of cyberbullying from time to time. Research by Theyazn H. H. Aldhyani et.al. has successfully classified comments in Instagram into bullying and non-bullying class using deep learning algorithms achieving best accuracy of 99%[8]. Another research conducted by Maria Ismiati in 2018 where negative comments on Instagram were detected using the Naïve Bayes algorithm with ±50 Instagram comments taken randomly as objects [9]. This research resulted in an accuracy value of 76.7%. Other research conducted by Luqyana, Cholissodin and Perdana in 2018 has conducted an analysis of Cyberbullying sentiment on Instagram comments using the SVM method which then produces an accuracy value of 90% [10]. Another study was conducted by Asep, Warih and Anisa who conducted sentiment analysis and summarizing product reviews using the Random Forest algorithm which resulted in an accuracy of 75% [11]. Another study conducted by Alhamda et al. in 2019 on sentiment analysis of cyberbullying in Instagram comments used the Naïve Bayes achieving an accuracy of 83.53%[12].

Another research also tackles similar problems in other social media platforms. Tahir, et.al. implemented an optimized Naïve Bayes to classify post in Social Media X yielding a satisfactory performance of 77.34% in accuracy[13]. Abdulwahab et.al. compares KNN, SVM, and Deep Learning methods to classify tweets into cyberbullying and non-cyberbullying class resulting in accuracy of 0.90 for KNN, 0.92 for SVM, and 0.96 for Deep Learning[14]. A group of researchers form Turkey compares different models in Deep Learning to detect cyberbullying in Turkish Twitter dataset[15]. Similarly, Handayani and Abas compares few Deep Learning models as well as varying the dataset size for classifying Twitter dataset of Indonesian population[16].

Numerous studies have compared the performance of Naïve Bayes, SVM, and Random Forest algorithms in text classification tasks. For instance, Pranckevičius and Marcinkevičius conducted research using Amazon review text data and found that Naïve Bayes slightly outperformed the other two algorithms, though the differences in performance were not substantial[17]. Conversely, Guia et al., utilizing Amazon mobile phone review data, demonstrated that SVM achieved the highest performance compared to the other algorithms[18]. These three algorithms are also often employed and compared with other methods in text classification tasks. For example, Ramachandran et al. used various machine learning algorithms, including Random Forest, for sentiment analysis of Instagram captions[19], while other studies focused on Twitter data[20], [21]. Nayak and Natarajan, using a Twitter movie review dataset, concluded that Naïve Bayes achieved the highest accuracy at 89%, with SVM and Random Forest following at 88% and 85%, respectively[21] . Ma'arif et al. utilized SVM and its optimization to analyze sentiment in investment app reviews from the Google Play Store dataset[22]. Given that no single classifier consistently outperforms others across all situations, it is essential to select algorithms based on the dataset's specific nature and characteristics, such as size, variance, and reliability.

The Support Vector Machine (SVM) is a machine learning algorithm applied in classification and regression tasks. It operates by using the principle of finding the optimal hyperplane that maximizes the margin between two classes. SVM is used for both binary classification (two classes) and multi-class classification. The SVM algorithm is highly effective for text categorization and can outperform the Naïve Bayes algorithm[23], [24].

The Naïve Bayes algorithm is a simple classification algorithm often used in many cases and yields good results. It works by finding the highest probability to classify test data into the correct category[9], [12].

Random Forest is an ensemble learning algorithm that uses a tree structure in its processes. In determining the class of data, Random Forest constructs decision trees by randomly selecting data. It employs a voting system based on the results of these decision trees. The method used by Random Forest makes predictions more efficient. Generally, Random Forest has advantages such as overcoming overfitting issues, being less sensitive to outliers, and having adjustable parameters.

The data utilized in this study is exclusively limited to Indonesian text. Several studies have analyzed Indonesian Instagram comments, particularly focusing on cyberbullying, but these studies often employ only a single algorithm and lack comparative analysis. For instance, Ramadhani et al. reported an accuracy of 84% using the Naïve Bayes algorithm with a dataset of 2,000 comments . Similarly, Naf'an et al. achieved 84% accuracy with Naïve Bayes using 455 comments[12]. Andriansyah et al. attained an accuracy of 79.4% using SVM with a dataset of 1,087 comments[25] , while Muhariya et al. reported an accuracy of 64.25% using the K-means algorithm[26].

3. METODE PENELITIAN

In this study three algorithm will be implemented namely SVM, Naive Bayes, and Random Forest. There were several stages that have been carried out, namely data collection, data preprocessing, data weighting, model building, and model testing. Figure 1 depicts the methodology used in this research.

3.1. DATA COLLECTION

In this study, the data was collected from three Instagram accounts between May and June 2020, covering posts from March to June 2020. The comments were categorized into three classes: negative, positive, and neutral. A total of 4,300 comments were analyzed. Table 1 presents the class distribution of the collected data.

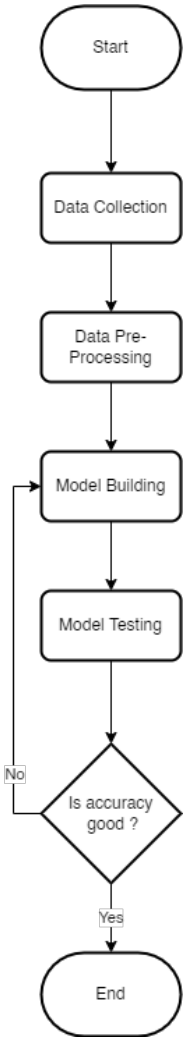

Figure 1 Research Methodology

Table 1 Data Class Distribution

Class	Amount
Negative	1524
Positive	*1544*
Neutral	1232

Table 2 depicts examples of comments and their assigned label. Each of the comment posts is manually labelled based on general perception of the text.

Table 2 Examples of comment and its assigned label

Number	Text	Class
1	Gamau dijelek2in tp elu sndr yg minta buat org jelek2in. manusia aneh ya elu itu	Negative
2	Sok cantik lu key.. gigi begitu juga ih.. Ga usah digitu2in knp si?? Pengen bgt jd bahan Bulian org.	Negative
3	Jangan lupa sholat ya key, berdo'a biar hidupmu bahagia dan jauh dari orang jahat dan yang suka memberi pengaruh buruk sama kamu	Positive
4	Jaga kesehatan key, istrht yg bagus pikiran yg bagus juga.. jgan terpengaruh dgan orang orang yg tdak baik.. hati hati key .berdoa jgan lupa tiap ba'da sholat	Positive
5	Key ajarin aku kuat seperti kamu dong	Neutral

3.2. DATA PREPROCESSING

After carrying out the data collection process, the next step is to do data preprocessing. The are several preprocessing stages, namely case folding, tokenization, filtering, and stemming.

Case Folding is the initial stage of text preprocessing which is used to change capital letters in text into lowercase. Besides that, it also removes characters other than letters such as punctuation marks, and others that are considered invalid, thereby reducing the noise in the text.

Tokenization involves breaking down text from sentences or paragraphs into smaller units, such as words. The process begins by transforming text data from sentence structures into individual word forms.

Text from Step 1	Tokenizing Result
sok cantik lu key gigi begitu juga ih ga usah digitu2in knp si pengen bgt jd bahan bulian org.	'sok', 'cantik', 'lu', 'key', 'gigi', 'begitu', 'juga', 'ih', 'ga', 'usah', 'digituin', 'knp', 'si', 'pengen',

	'bgt', 'jd', 'bahan', 'bulian', 'org'

Filtering is an important process of taking words from tokenizing results. In the filtering process using a stopword. Stopwords are words that are not descriptive and are not important words in a document so they can be omitted like the words "which", "and", "in", "from" and others.

Text from Step 2	Filtering Result
sok cantik lu key gigi begitu juga ih ga usah digituin knp si pengen bgt jd bahan bulian org	cantik gigi begitu juga pengen bahan

Stemming is the process of identifying the root or base words derived from the filtering results. This process utilizes a lexical library to aid in transforming Indonesian affixes into their base forms.

Text before Stemming	Stemming Result
jangan lupa dengan kebersihan kuku	Jangan lupa dengan bersih kuku

3.3. DATA WEIGHTING

After completing the text data preprocessing stage, the next step is data weighting, which involves converting textual data into numerical form. The data weighting process uses the TF-IDF method. TF-IDF stands for "Term Frequency-Inverse Document Frequency." This method is a method for calculating a score for each word (term) that appears in a document. It is a popular weighting method used in information retrieval to assess the relevance of a document to a query.

The TF-IDF weighting method assigns a score to each term in a document. The TF-IDF score for a term in a document is calculated by multiplying its term frequency (TF) by its inverse document frequency (IDF), with a higher score indicating greater importance.

Cosine similarity is a method used to measure the similarity between a query and a document by calculating the cosine of the angle between their vectors in a high-dimensional space. The process involves converting the query and document into vectors representing TF-IDF scores, computing the dot product of these vectors, calculating their magnitudes, and then dividing the dot product by the product of the magnitudes to obtain the cosine similarity. This process is repeated for each document in the corpus, ranking them in descending order of similarity to the query. The most similar documents are presented to the user as the most relevant.

In classification models, the TF-IDF score is crucial for representing text data numerically. The TF-IDF score helps create a feature matrix, where each row corresponds to a document and each column to a unique term. In this matrix, each cell represents the TF-IDF score of a term within a document. This matrix serves as input for machine learning algorithms, which use it to learn patterns in the data and predict labels or categories for new documents. The TF-IDF score highlights the relative importance of each term within a document and across the corpus, identifying key features crucial for classification. A low TF-IDF score indicates that a term frequently appears in a document, whereas a high score suggests that a term appears infrequently, making it significant. The higher the TF-IDF score, the greater the similarity to the term. Table 3 provides an example of TF-IDF computation and its results.

Table 3 Example of TF-IDF Computation Result

Query	TF	IDF	TF-IDF
jijik	4	0.862728	3.450912
lewat	1	1.792749	1.792749
gigi	2	1.271833	2.543666
suka	1	1.963744	1.963744
anjing	2	1.374692	2.749384
cantik	1	2.173649	2.173649
gila	3	0.963821	2.891463

Inverse Document Frequency (IDF) indicates the relationship between the availability of a term within a document. The smaller the Term Frequency (TF) occurrence, the larger the IDF value. The TF-IDF algorithm is used to combine sentences with a set of documents, and the most common method for calculating TF-IDF values involves multiplying the TF value by the IDF value to obtain the TF-IDF weight.

3.4. BUILDING MODELS

The next step is to build the model, formally known as the training stage, where the algorithm is run using the data as input and later evaluated with the testing data. In this stage, the data is divided into training and testing sets, with 20% of the data allocated for testing as shown in Table 4. Additionally, parameter tuning can be performed using grid search to identify the optimal parameters for the algorithm, thereby enhancing its performance.

Table 4 Data Split

	Percentage	Number of posts
Training data	80%	3440
Testing data	20%	860

A grid search is employed to optimize parameter tuning, with the goal of enhancing the algorithm's performance. In this study, parameter tuning was conducted for the SVM and Random Forest algorithms, each of which has distinct parameters that require optimization.

Support Vector Machine (SVM) parameters include several key components: the kernel , the C parameter, gamma , the degree parameter (relevant only for the polynomial kernel). In this study, grid search tuning identified optimal parameters for the SVM algorithm: C = 10, gamma = "auto," and a linear kernel function.

Random Forest parameters also include several critical components: the n_estimators , the max_features parameter , the max_depth parameter, the min_samples_split parameter , the min_samples_leaf parameter, the bootstrap parameter and the class_weight parameter. Proper tuning can help achieve a balance between bias and variance, leading to better generalization on unseen data. The grid search parameter tuning for the Random Forest algorithm conducted in this study identified the optimal parameters: criterion set to "gini," max_depth to 8, max_features to "auto," and n_estimators to 500.

3.5. MODEL TESTING

The testing process is carried out to measure the accuracy of the results of each model that has been proposed. This performance measurement uses the accuracy, precision, recall, and F1 score values calculation which aims to determine the differences in the performance of the SVM, Naïve Bayes and Random Forest algorithms in the classification of sentiment on Instagram comments. Measurement of accuracy value, precision value, recall value is processed using a 3x3 confusion matrix.

Table 5 Confusion Matrix

	Prediction: Positive (P)	Prediction: Negative (N)	Prediction: Neutral (NR)
Actual: Positive (P)	*PP*	*PN*	*PNR*
Actual: Negative (N)	*NP*	*NN*	*NNR*
Actual: Neutral (NR)	*NRP*	*NRN*	*NRNR*

From Table 5 we can formulate the evaluation measure as follow:

3.5.1 Accuracy

$$accuracy = \frac{PP + NN + NRNR}{PP + PN + PNR + NP + NN + NNR + NRP + NRN + NRNR} * 100\%$$

3.5.2 Precision

$$Precision_{positive} = \frac{PP}{PP+NP+NRP}$$

$$Precision_{negative} = \frac{NN}{PN+NN+NRN}$$

$$Precision_{Neutral} = \frac{NRNR}{PNR+NNR+NRNR}$$

3.5.3 Recall

$$Recall_{positive} = \frac{PP}{PP+PN+PNR}$$

$$Recall_{negative} = \frac{NN}{NP+NN+NNR}$$

$$Recall_{Neutral} = \frac{NR}{NRP+NRN+NRNR}$$

3.5.4. F1 Score

$$F1score_{positif} = \frac{2 * Precision_{positif} * Recall_{positif}}{Precision_{positif} + Recall_{positif}}$$

$$F1score_{negatif} = \frac{2 * Precision_{negatif} * Recall_{negatif}}{Precision_{negatif} + Recall_{negatif}}$$

$$F1score_{netral} = \frac{2 * Precision_{netral} * Recall_{netral}}{Precision_{netral} + Recall_{netral}}$$

4. RESULT AND DISCUSSION

The three algorithms were implemented and evaluated using the testing data, resulting in a confusion matrix for each algorithm. Based on these confusion matrices, evaluation measures were computed using the formulas provided in Sections 3.5.1 to 3.5.4. The performance metrics for each algorithm were calculated, resulting in a comparison of the average values, as presented in Table 6.

Table 6 Performance Result

	SVM	Random forest	Naive bayes	Random forest with *grid search*	SVM with *grid search*
Accuracy	75.47%	69.88%	75.47%	66.51%	97.79%
Average Precision	76.13%	70.05%	78.11%	81.18%	97.81%
Average Recall	75.47%	69.88%	75.47%	66.51%	97.79%
Average F1	75.55%	69.81%	76.02%	63.79%	97.79%

Table 6 indicates that the SVM algorithm with parameter tuning achieves the highest performance. This is due to the parameter tuning process resulting in a linear kernel function, which operates optimally for binary classification tasks. The SVM algorithm also performs effectively in multiclass classification scenarios, such as text mining, when utilizing a linear kernel, as demonstrated in this study. This indicates that the grid search process for SVM successfully identified optimal parameters, thereby achieving high accuracy. Consequently, SVM with parameter tuning is deemed the most effective algorithm in this study.

Following the evaluation of each algorithm's performance, the next step involved developing a prototype to classify comments whether it contains cyberbullying sentiment. The prototype implemented SVM algorithm and was developed using the Python programming language and the Jupyter Notebook software.

Figure 2-5 depicts the prototypes that are developed to test the model for new comments. Figure 3 displays examples of positive comment and the outcome of the test while Figures 4 and 5 display the prototype when provided with neutral and negative comments.

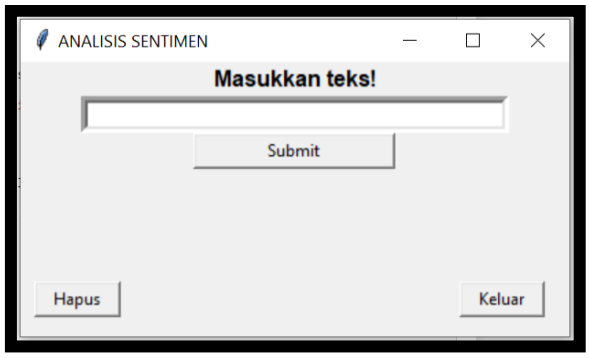

Figure 2 User Interface of Testing Prototype

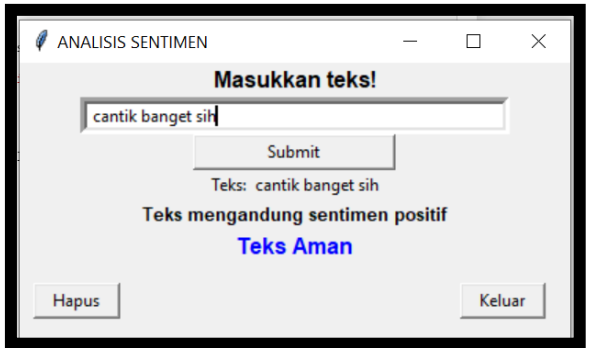

Figure 3 Testing with Positive Comment

Figure 4 Testing with Neutral Comment

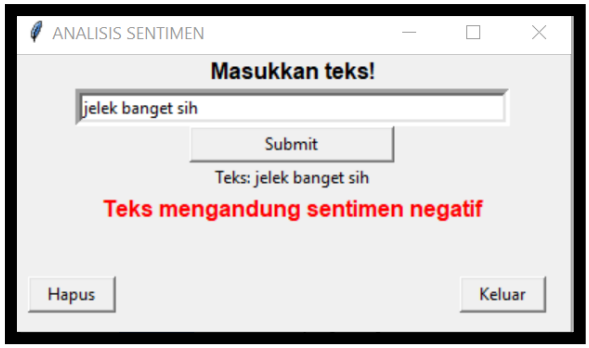

Figure 5 Testing with Negative Comment

5. CONCLUSION

Based on the research findings, it can be concluded that the Support Vector Machine (SVM) algorithm with parameter tuning demonstrated the

highest performance, achieving an accuracy of 91.13%, an average precision of 91.41%, an average recall of 91.13%, and an average F1 score of 91.03%. The application of parameter tuning successfully enhanced the performance of each algorithm tested. These results represent an improvement over previous studies where SVM was utilized. The model developed within the prototype effectively distinguishes between positive, neutral, and negative comments.

ACKNOWLEDGEMENT

We gratefully acknowledge the financial support from [University Name] for this publication. Their funding has been crucial to the success of our research.

REFERENCES

[1] Stacy Jo Dixon, 'Leading countries based on Instagram audience size as of January 2024 (in millions)', 2024.

[2] Ditch the label, 'The annual bullying survey', 2017.

[3] G. K. Bhatia, 'The role of education in combatting cyberbullying in Indonesia', *Chilfund Alliance*, 2023. [Online]. Available: https://childfundalliance.org/2023/11/28/the-role-of-education-in-combatting-cyberbullying-in-indonesia/.

[4] H. Dani, J. Li, and H. Liu, 'Sentiment Informed Cyberbullying Detection in Social Media', *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10534 LNAI, pp. 52–67, 2017.

[5] J. Van Tiel, 'Cyberbullying, an overlooked and ever growing danger to the development of children A Report from KidsRights', 2020.

[6] A. Y. Laora and F. Sanjaya, 'Fenomena Cyberbullying di Media Sosial Instagram (Studi Deskriptif Tentang Kesehatan Mental Pada Generasi Z Usia 20-25 Tahun di Jakarta)', *Oratio Directa*, vol. 3, no. 1, pp. 346–368, 2021.

[7] A. Rachmayanti and Y. Candrasari, 'Perilaku Cyberbullying Di Instagram', *Linimasa J. Ilmu Komun.*, vol. 5, no. 1, pp. 1–12, 2022.

[8] T. H. H. Aldhyani, M. H. Al-Adhaileh, and S. N. Alsubari, 'Cyberbullying Identification System Based Deep Learning Algorithms', *Electron.*, vol. 11, no. 20, 2022.

[9] M. B. Ismiati, 'Deteksi Komentar Negatif Di Instagram Menggunakan Algoritma Naive Bayes Calssifier', in *Prosiding SNST ke-9*, 2018, vol. 9, pp. 243–248.

[10] W. Athira Luqyana, I. Cholissodin, and R. S. Perdana, 'Analisis Sentimen Cyberbullying pada Komentar Instagram dengan Metode Klasifikasi Support Vector Machine', *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 4704–4713, 2018.

[11] A. Aprianto, W. Maharani, and A. Herdiani, 'Analisis Sentimen Dan Peringkasan Opini Pada Ulasan Produk Menggunakan Algoritma Random Forest', *eProceedings Eng.*, vol. 3, no. 3, 2016.

[12] M. Z. Naf'an, A. A. Bimantara, A. Larasati, E. M. Risondang, and N. A. S. Nugraha, 'Sentiment Analysis of Cyberbullying on Instagram User Comments', *J. Data Sci. Its Appl.*, vol. 2, no. 1, pp. 88–98, 2019.

[13] S. F. Tahir and C. A. Sugianto, 'Optimasi Naive Bayes Menggunakan Algoritma Genetika Pada Klasifikasi Komentar Cyberbullying Pada Media Sosial X', *J. Inform. dan Tek. Elektro Terap.*, vol. 12, 2024.

[14] A. Alabdulwahab, M. A. Haq, and M. Alshehri, 'Cyberbullying Detection using Machine Learning and Deep Learning', *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 10, pp. 424–432, 2023.

[15] Ç. O. Aliyeva and M. Yağanoğlu, 'Deep learning approach to detect cyberbullying on twitter', *Multimed. Tools Appl.*, 2024.

[16] T. P. Handayani and M. I. Abas, 'Comparative Analysis of CNN-LSTM and LSTM Models for Cyberbullying Detection with Increasing Dataset Sizes', *J. Ilmu Komput.*, vol. 4, no. 2, pp. 75–85, 2024.

[17] T. Pranckevičius and V. Marcinkevičius, 'Comparison of Naive Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification', *Balt. J. Mod. Comput.*, vol. 5, no. 2, 2017.

[18] M. Guia, R. R. Silva, and J. Bernardino, 'Comparison of Naive Bayes, support vector machine, decision trees and random forest on sentiment analysis', *IC3K 2019 - Proc. 11th Int. Jt. Conf. Knowl. Discov. Knowl. Eng. Knowl. Manag.*, vol. 1, pp. 525–531, 2019.

[19] A. Ramachandran, S. Ashok, and T. Remya Nair, 'A Factual Sentiment Analysis on Instagram Data - A Comparative Study Using Machine Learning Algorithms', *Proc. - 2022 Algorithms, Comput. Math. Conf. ACM 2022*, pp. 1–5, 2022.

[20] J. Singh and P. Tripathi, 'Sentiment analysis of Twitter data by making use of SVM, Random Forest and Decision Tree algorithm', *Proc. - 2021 IEEE 10th Int. Conf. Commun. Syst. Netw. Technol. CSNT 2021*, pp. 193–198, 2021.

[21] A. Nayak and S. Natarajan, 'Comparative

study of Naïve Bayes , Support Vector Machine and Random Forest Classifiers in Sentiment Analysis of Twitter feeds', *Int. J. Adv. Stud. Comput. Sci. Eng. IJASCSE*, vol. 5, no. 1, pp. 14–17, 2016.

[22] M. S. Ma'arif, J. H. Jaman, and A. S. Y. Irawan, 'Analisis Sentimen Ulasan Aplikasi Investasi Menggunakan Algoritma Support Vector Machine Berbasis Particle Swarm Optimization', *J. Inform. dan Elektro Terap.*, 2024.

[23] F. Millennianita, U. Athiyah, and A. W. Muhammad, 'Comparison of Naïve Bayes Classifier and Support Vector Machine Methods for Sentiment Classification of Responses to Bullying Cases on Twitter', vol. 1, no. 1, pp. 11–26, 2024.

[24] T. L. Nikmah, M. Z. Ammar, Y. R. Allatif, R. M. P. Husna, P. A. Kurniasari, and A. S. Bahri, 'Comparison of LSTM, SVM, and naive bayes for classifying sexual harassment tweets', *J. Soft Comput. Explor.*, vol. 3, no. 2, 2022.

[25] M. Andriansyah *et al.*, 'Cyberbullying comment classification on Indonesian Selebgram using support vector machine method', *Proc. 2nd Int. Conf. Informatics Comput. ICIC 2017*, vol. 2018-January, pp. 1–5, 2017.

[26] A. Muhariya, I. Riadi, Y. Prayudi, and I. A. Saputro, 'Utilizing K-means Clustering for the Detection of Cyberbullying Within Instagram Comments', *Ing. des Syst. d'Information*, vol. 28, no. 4, pp. 939–949, 2023.